A relaxation method with projective integration for solving nonlinear systems of hyperbolic conservation laws

Pauline Lafitte
Ward Melis
Giovanni Samaey

Report TW 649, June 2014

KU Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

A relaxation method with projective integration for solving nonlinear systems of hyperbolic conservation laws

Pauline Lafitte[*]
Ward Melis
Giovanni Samaey

Report TW 649, June 2014

Department of Computer Science, KU Leuven

Abstract

We present a general, high-order, fully explicit, relaxation scheme for systems of nonlinear hyperbolic conservation laws in multiple dimensions. The scheme consists of two steps: at first, the nonlinear hyperbolic conservation law is approximated through a kinetic equation with BGK source term. Then, this kinetic equation is integrated using a projective integration method, which first takes a few small (inner) steps with a simple, explicit method (such as direct forward Euler) to damp out the stiff components of the solution, after which the time derivative is estimated and used in an (outer) Runge-Kutta method of arbitrary order. We show that, with an appropriate choice of inner step size, the time step restriction on the outer time step is similar to the CFL condition for the hyperbolic conservation law. Moreover, the number of inner time steps is also independent of the scaling parameter. We analyze stability and consistency, and illustrate with numerical results (linear advection, Burgers' equation and the Euler equations) in one and two spatial dimensions.

Keywords : kinetic equations, projective integration, asymptotic-preserving, hyperbolic conservation laws, higher-order.

[*]Laboratoire de Mathématiques Appliquées aux Systèmes, Ecole Centrale Paris, Grande Voie des Vignes, 92290 Châtenay-Malabry, France (pauline.lafitte@ecp.fr).

A relaxation method with projective integration for solving nonlinear systems of hyperbolic conservation laws

Pauline Lafitte * Ward Melis† Giovanni Samaey‡

June 12, 2014

Abstract

We present a general, high-order, fully explicit, relaxation scheme for systems of nonlinear hyperbolic conservation laws in multiple dimensions. The scheme consists of two steps: at first, the nonlinear hyperbolic conservation law is approximated through a kinetic equation with BGK source term. Then, this kinetic equation is integrated using a projective integration method, which first takes a few small (inner) steps with a simple, explicit method (such as direct forward Euler) to damp out the stiff components of the solution, after which the time derivative is estimated and used in an (outer) Runge-Kutta method of arbitrary order. We show that, with an appropriate choice of inner step size, the time step restriction on the outer time step is similar to the CFL condition for the hyperbolic conservation law. Moreover, the number of inner time steps is also independent of the scaling parameter. We analyze stability and consistency, and illustrate with numerical results (linear advection, Burgers' equation and the Euler equations) in one and two spatial dimensions.

1 Introduction

Hyperbolic conservation laws arise in numerous physical applications, such as fluid dynamics, plasma physics, traffic modeling and electromagnetism (see, e.g., [19, 31]). They express the conservation of physical quantities (such as mass, momentum, or energy) and may be supplemented with boundary conditions that control influx or outflux at the boundaries of the physical domain [19]. In this paper, we consider a system of hyperbolic conservation laws in multiple spatial dimensions:

$$\partial_t \mathbf{u} + \nabla_{\mathbf{x}} \cdot \mathbf{F}(\mathbf{u}) = 0, \tag{1}$$

or, equivalently,

$$\partial_t \mathbf{u} + \sum_{d=1}^{D} \partial_{x^d} \mathbf{F}^d(\mathbf{u}) = 0, \tag{2}$$

in which $\mathbf{x} = (x^d)_{d=1}^D \in \mathbb{R}^D$ represents the space variables (D being the number of spatial dimensions), $\mathbf{u}(\mathbf{x}, t) := (u_m(\mathbf{x}, t))_{m\in\{1,\dots,M\}} \in \mathbb{R}^M$ represents the conserved quantities, and $\mathbf{F}(\mathbf{u}) \in \mathbb{R}^{M\times D}$ represents the flux functions.

Hyperbolic conservation laws are often solved using a finite volume method [19, 22], which is derived from the integral expression of the conservation law. To this end, in a scalar one-dimensional setting and with a spatially uniform grid, the domain is divided in I cells $\mathcal{C}_i = [x_{i-1/2}, x_{i+1/2}]$ with constant cell width Δx over which the cell average of the solution $u(x, t)$ to the conservation law

$$\partial_t u + \partial_x F(u) = 0, \tag{3}$$

*Laboratoire de Mathématiques Appliquées aux Systèmes, Ecole Centrale Paris, Grande Voie des Vignes, 92290 Châtenay-Malabry, France (pauline.lafitte@ecp.fr).

†Department of Computer Science, K.U. Leuven, Celestijnenlaan 200A, 3001 Leuven, Belgium (ward.melis@cs.kuleuven.be).

‡Department of Computer Science, K.U. Leuven, Celestijnenlaan 200A, 3001 Leuven, Belgium (giovanni.samaey@cs.kuleuven.be).

is approximated at time $t = t^n$ by

$$U_i^n \approx \frac{1}{\Delta x} \int_{\mathcal{C}_i} u(x, t^n) dx. \tag{4}$$

Note that boldface is removed whenever the quantities are scalar. A numerical scheme is then constructed by integrating the conservation law (3) in space over the cell $\mathcal{C}_i$ and in time from t^n to t^{n+1} to obtain

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{\Delta x} \left(F_{i+1/2}^n - F_{i-1/2}^n \right), \tag{5}$$

in which $\Delta t = t^{n+1} - t^n$ and the *numerical flux* satisfies

$$F_{i\pm1/2}^n \approx \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} F\left(u\left(x_{i\pm1/2}, t\right)\right) dt. \tag{6}$$

Clearly, equation (5) is conservative by construction. The numerical fluxes $F_{i\pm1/2}^n$ can be obtained by constructing an (approximate) Riemann solver, based on a (possibly high-order) reconstruction of the solution in each of the cells using interpolation over the neighboring cells [19, 24].

To avoid the (possibly tedious) computation of the solutions of local Riemann problems, relaxation methods, see, e.g., [1, 12, 13, 20] offer an interesting alternative. In a relaxation method, the conservation law (1) is approximated by a problem of higher dimension containing a small relaxation parameter ϵ such that, when ϵ tends to zero, the original problem is recovered. The idea is that some of the difficulties associated with the original problem are avoided, while, for sufficiently small ϵ, the relaxation problem is a good approximation of the problem of interest. In this paper, we will consider the relaxation problem to be a kinetic BGK equation: a mesoscopic problem is introduced to offer a better description of the distribution of the particles in terms of time, space and velocity variables. In a scalar one-dimensional setting, this equation describes the evolution of a distribution function $f(x, v, t)$ of particles at position x with velocity v at time t and takes the following form:

$$\partial_t f^\epsilon + v \partial_x f^\epsilon = \frac{1}{\epsilon} \left(\mathcal{M}_v \left(u^\epsilon \right) - f^\epsilon \right). \tag{7}$$

The left hand side of equation (7) describes the transport of the particles whereas the right hand side represents the collisions between particles, which is modeled as a linear relaxation to the Maxwellian $\mathcal{M}_v \left(u^\epsilon \right)$ with a relaxation time ϵ. The advantage of the kinetic equation (7) over the conservation law (3) is the fact that the advection term in (7) is now linear; the disadvantage is the appearance of a stiff source term, which requires special care during time integration. The first methods, proposed in [1, 13] are based on splitting, thus restricting the order in time to 2. More recently, several *asymptotic-preserving* methods based on IMEX techniques (in the sense of Jin [11]) have been proposed that integrate the Boltzmann equation in the hyperbolic and diffusive regimes with a computational cost that is independent of ϵ (see [8] and references within). An appealing idea along this line of thought, based on IMEX Runge-Kutta methods, is presented in [4]. Unfortunately, the proposed method is not very robust since it breaks down in intermediate regimes. An improvement was proposed in [6].

A robust and fully explicit alternative to splitting and IMEX, which allows for time integration of stiff systems with arbitrary order of accuracy in time is projective integration. Projective integration was proposed in [9] for stiff systems of ordinary differential equations. In such stiff problems, the fast modes, corresponding to the Jacobian eigenvalues with large negative real parts, decay quickly, whereas the slow modes correspond to eigenvalues of smaller magnitude and are the solution components of practical interest. Projective integration allows a stable yet explicit integration of such problems by first taking a few small (inner) steps with a simple, explicit method, until the transients corresponding to the fast modes have died out, and subsequently projecting (extrapolating) the solution forward in time over a large (outer) time step. In [17], projective integration was analyzed for kinetic equations with a diffusive scaling. An arbitrary order version, based on Runge-Kutta methods, has been proposed recently in [16], where it was also analyzed for kinetic equations with an advection-diffusion limit. These methods fit within recent research efforts on numerical methods for multiscale simulation [14, 15, 29, 30]; see also [7, 26, 27] for related approaches. Alternative approaches to obtain a higher-order projective integration scheme have been proposed in [18, 23].

In this paper, we construct a relaxation method with projective integration to simulate any hyperbolic system of conservation laws in multiple space dimensions. The resulting scheme turns out to be

fully explicit, of arbitrary order in time and fully general, avoiding the construction of complicated approximate Riemann solvers. High-order projective Runge-Kutta methods are proposed and analyzed in the companion paper [16]. The present paper focuses solely on their use in combination with a relaxation method, including ample numerical illustrations.

This paper is structured as follows. In section 2, we will introduce the kinetic equations that will form the basis of the relaxation method, and discuss their asymptotic equivalence with the original hyperbolic problem. In section 3, we describe the projective integration method that will be used to integrate these kinetic equations. We then briefly review the convergence results obtained in [16] in section 4, which we can use to determine the method parameters for projective integration. The main results of the paper are reported in section 5, where the projective integration methods will be applied to a set of benchmark problems in both one and two space dimensions: linear advection, nonlinear conservation and Sod's shock test. We conclude in section 6 with a brief discussion and some ideas for future work.

2 Relaxation systems

2.1 Kinetic equation and hydrodynamic limit

To solve equation (1), we introduce, as in [1], the (hyperbolically scaled) kinetic equation

$$\partial_t \mathbf{f}^\epsilon + \mathbf{v} \cdot \nabla_{\mathbf{x}} \cdot \mathbf{f}^\epsilon = \frac{1}{\epsilon} \left(\mathcal{M}_v \left(\mathbf{u}^\epsilon \right) - \mathbf{f}^\epsilon \right), \tag{8}$$

or, equivalently,

$$\partial_t \mathbf{f}^\epsilon + \sum_{d=1}^{D} v^d \partial_{x^d} \mathbf{f}^\epsilon = \frac{1}{\epsilon} \left(\mathcal{M}_v \left(\mathbf{u}^\epsilon \right) - \mathbf{f}^\epsilon \right), \tag{9}$$

modeling the evolution of a vector of particle distribution functions $\mathbf{f}^\epsilon(\mathbf{x}, \mathbf{v}, t) = (f_m^\epsilon(\mathbf{x}, \mathbf{v}, t))_{m=1}^M \in \mathbb{R}^M$. The particle positions and velocities are represented as $\mathbf{x} = (x^d)_{d=1}^D \in \mathbb{R}^D$ and $\mathbf{v} = (v^d)_{d=1}^D \in V \subset \mathbb{R}^D$ respectively, and the right hand side of (9) represents a BGK collision operator [2], modeling linear relaxation of $\mathbf{f}^\epsilon$ to a Maxwellian distribution $\mathcal{M}_v(\mathbf{u}^\epsilon) \in \mathbb{R}^M$, in which $\mathbf{u}^\epsilon(\mathbf{x}, t) = \langle \mathbf{f}^\epsilon(\mathbf{x}, \mathbf{v}, t) \rangle$ is the density, obtained via averaging over the measured velocity space (V, μ),

$$\mathbf{u} := \langle \mathbf{f} \rangle = \int_V \mathbf{f} d\mu(\mathbf{v}). \tag{10}$$

The advantage of this kinetic formulation is that the advection term is now linear, and therefore easier to discretize. The disadvantage is the increased dimension, as well as the introduction of the stiff source term of size $O(1/\epsilon)$. The projective integration scheme that we will propose in section 3 allows to integrate this stiff source term using an explicit method of arbitrary order.

To ensure that the kinetic equation (9) converges to the conservation law (1) in the hydrodynamic limit $\epsilon \to 0$, one requires

$$\begin{cases} \langle \mathcal{M}_v \left(\mathbf{u} \right) \rangle = \mathbf{u}, \\ \langle v^d \mathcal{M}_v \left(\mathbf{u} \right) \rangle = \mathbf{F}^d(\mathbf{u}), \qquad 1 \leq d \leq D. \end{cases} \tag{11}$$

Then, one can show [1] that, in the limit of $\epsilon \to 0$, the kinetic model (9) is approximated by the following equation:

$$\partial_t \mathbf{u}^\epsilon + \nabla_{\mathbf{x}} \cdot \mathbf{F} \left(\mathbf{u}^\epsilon \right) = \epsilon \, \nabla_{\mathbf{x}} \cdot \left(\mathbf{B} \, \nabla_{\mathbf{x}} \mathbf{u}^\epsilon \right), \tag{12}$$

or, equivalently,

$$\partial_t \mathbf{u}^\epsilon + \sum_{d=1}^{D} \partial_{x^d} \mathbf{F}^d \left(\mathbf{u}^\epsilon \right) = \epsilon \sum_{d=1}^{D} \partial_{x^d} \left(\sum_{d'=1}^{D} \mathbf{B}_{dd'} \, \partial_{x^{d'}} \mathbf{u}^\epsilon \right), \tag{13}$$

with the diffusion matrix $\mathbf{B}$ given as

$$\mathbf{B}_{dd'}(\mathbf{u}) := \langle v^d v^{d'} \partial_{\mathbf{u}} \mathcal{M}_v(\mathbf{u}) \rangle - \partial_{\mathbf{u}} \mathbf{F}^d \partial_{\mathbf{u}} \mathbf{F}^{d'}, \tag{14}$$

in which the $M \times M$ matrices $\partial_{\mathbf{u}} \mathcal{M}_v(\mathbf{u})$ and $\partial_{\mathbf{u}} \mathbf{F}^d$ represent the Jacobian matrices of $\mathcal{M}_v(\mathbf{u})$ and $\mathbf{F}(\mathbf{u})$ respectively.

Clearly, equation (9) is consistent with equation (1) to order 1 in ϵ. Moreover, the analysis reveals an additional condition on $\mathcal{M}$ and V. Indeed, to ensure the parabolicity of (13), the diffusion matrix $\mathbf{B}$ should be positive definite. This leads to the so-called *subcharacteristic condition* [1,5],

$$\sum_{d,d'=1}^{D} \left(\mathbf{B}_{dd'}(\mathbf{u}) \xi^{d'} \cdot \xi^{d} \right) \geq 0, \tag{15}$$

for all ξ^d, $1 \leq d \leq D$ in $\mathbb{R}^M$.

In what follows, we will always assume that the velocity space is discrete and of the form

$$V := \{\mathbf{v}_j\}_{j=1}^{J}, \qquad d\mu(\mathbf{v}) = \sum_{j=1}^{J} w_j \delta(\mathbf{v} - \mathbf{v}_j), \tag{16}$$

with $\mathbf{v}_j$ denoting the chosen velocities and w_j the corresponding weights. Due to this choice of V the kinetic equation (8) breaks up into a system of J coupled partial differential equations,

$$\partial_t \mathbf{f}_j^\epsilon + \mathbf{v}_j \cdot \nabla_{\mathbf{x}} \mathbf{f}_j^\epsilon = \frac{1}{\epsilon} \left(\mathcal{M}_j \left(\mathbf{u}^\epsilon \right) - \mathbf{f}_j^\epsilon \right), \qquad 1 \leq j \leq J, \tag{17}$$

in which $\mathbf{f}_j^\epsilon(x,t) \equiv \mathbf{f}^\epsilon(x, v_j, t)$, and the only coupling between different velocities is through the computation of $\mathbf{u}^\epsilon$. As $\epsilon \to 0$, a Chapman-Enskog expansion allows to write

$$\mathbf{f}_j^\epsilon = \mathcal{M}_j \left(\mathbf{u}^\epsilon \right) + O(\epsilon) \tag{18}$$

so that, injecting it in (17) and taking the mean value over V, we get

$$\partial_t \langle \mathcal{M}_j \left(\mathbf{u}^\epsilon \right) \rangle + \nabla_{\mathbf{x}} \langle \mathbf{v}_j \cdot \mathcal{M}_j \left(\mathbf{u}^\epsilon \right) \rangle = O(\epsilon).$$

Finally, the compatibility conditions (11) imply

$$\partial_t \mathbf{u}^\epsilon + \sum_{d=1}^{D} \partial_{x_d} F^d(\mathbf{u}^\epsilon) = O(\epsilon). \tag{19}$$

2.2 One-dimensional examples

In one space dimension, we write equation (1) as

$$\partial_t \mathbf{u} + \partial_x \mathbf{F} \left(\mathbf{u} \right) = 0, \tag{20}$$

in which $t \geq 0$ (resp. $x \in \mathbb{R}$) represents the time (resp. space) variable, $\mathbf{u}(x,t) = (u_m(x,t))_{m=1}^M \in \mathbb{R}^M$ embodies the conserved quantities, and $\mathbf{F}(\mathbf{u}) = (F_m(\mathbf{u}))_{m=1}^M \in \mathbb{R}^M$ represents the flux functions. Correspondingly, the kinetic equation (9) becomes

$$\partial_t \mathbf{f}^\epsilon + v \partial_x \mathbf{f}^\epsilon = \frac{1}{\epsilon} \left(\mathcal{M}_v \left(\mathbf{u}^\epsilon \right) - \mathbf{f}^\epsilon \right), \tag{21}$$

with the particle distribution function $\mathbf{f}^\epsilon(x,v,t) = (f_m^\epsilon(x,v,t))_{m=1}^M \in \mathbb{R}^M$, and the particle velocities represented as $v \in V \subset \mathbb{R}$.

We choose a discrete measured velocity space with an even number J of velocities that satisfy $v_{J-j+1} \equiv -v_j$, and a Maxwellian of the form

$$\mathcal{M}_v(\mathbf{u}^\epsilon) = \mathbf{u}^\epsilon + \frac{\mathbf{F}(\mathbf{u}^\epsilon)}{v}. \tag{22}$$

With these choices, the conditions (11) are clearly satisfied. The specific values of the velocities v_j need to be chosen such that the subcharacteristic condition (15) is satisfied. When we further restrict to a scalar case, i.e., $M = 1$,

$$\partial_t u + \partial_x F \left(u \right) = 0, \tag{23}$$

it can be checked that the subcharacteristic condition is always satisfied as soon as

$$1 + \partial_u F(u)/v_j \geq 0, \quad \text{or, equivalently,} \quad |v_j| \geq |\partial_u F(u)| \quad 1 \leq j \leq J. \tag{24}$$

Note again that all boldfaced typesetting is removed for a scalar case.

For the numerical illustrations, we choose concretely the following examples:

Example 2.1 *The linear scalar advection equation,*

$$F(u) = a \cdot u, \qquad a \in \mathbb{R}. \tag{25}$$

Example 2.2 *The scalar Burgers' equation,*

$$F(u) = u^2/2. \tag{26}$$

Example 2.3 *The one-dimensional Euler equations*

$$\begin{aligned} \mathbf{u} &= (\rho, \rho\bar{v}, E), & (27)\\ \mathbf{F}(\mathbf{u}) &= (\rho\bar{v}, \rho\bar{v}^2 + P, E + P\bar{v}), & (28) \end{aligned}$$

with the equation of state

$$P = (\gamma - 1)\left(E - \rho\bar{v}^2\right). \tag{29}$$

2.3 Two-dimensional examples

In two space dimensions, we write equation (1) as

$$\partial_t \mathbf{u} + \partial_x \mathbf{F}^x(\mathbf{u}) + \partial_y \mathbf{F}^y(\mathbf{u}) = 0, \tag{30}$$

where $t \geq 0$ (resp. $x, y \in \mathbb{R}$) represents the time (resp. space) variables $\mathbf{u}(x, y, t) = (u_m(x, y, t))_{m=1}^M \in \mathbb{R}^M$ represents the conserved quantities, and $\mathbf{F}^{x,y}(\mathbf{u}) = (F_m^{x,y}(\mathbf{u}))_{m=1}^M \in \mathbb{R}^M$ represents the fluxes in the x and y direction respectively. Correspondingly, the kinetic equation (9) becomes

$$\partial_t \mathbf{f}^\epsilon + v^x \partial_x \mathbf{f}^\epsilon + v^y \partial_y \mathbf{f}^\epsilon = \frac{1}{\epsilon}\left(\mathcal{M}_v\left(\mathbf{u}^\epsilon\right) - \mathbf{f}^\epsilon\right), \tag{31}$$

in which the particle distribution function $\mathbf{f}^\epsilon(x, y, v^x, v^y, t) = (f_m^\epsilon(x, y, v^x, v^y, t))_{m=1}^M \in \mathbb{R}^M$ and the particle velocities $\mathbf{v} = (v^x, v^y) \in V \subset \mathbb{R}^2$, with $v^{x,y}$ the velocity of the particles in the x and y direction respectively.

Compared to the one-dimensional setting, the choice of the Maxwellian and the description of the discrete velocity space are considerably more elaborate, and many options have been documented, see, e.g., [1, 3, 21]. In the numerical examples in this paper, we choose the orthogonal velocities method, see, e.g., [1], which we now detail for the scalar case ($M = 1$). In this method, we choose a set of velocities with varying length and direction. Specifically, we first fix a maximal velocity length $v_{\max}$. We then consider R different velocity lengths

$$\rho_r = \frac{r}{R} v_{\max}, \qquad 1 \leq r \leq R,$$

and $4S$ different velocity directions

$$\theta_s = \frac{s\,\pi}{2S}, \qquad 1 \leq s \leq 4S,$$

with $R, S \geq 1$. We then obtain $J = 4RS$ velocities $\mathbf{v}_j = (v_j^x, v_j^y)$, $1 \leq j \leq J$, by assigning an index $j = (r-1)\,4S + s$ to every length-direction pair (r, s), and writing

$$v_j^x = \rho_r \cos(\theta_s), \qquad v_j^y = \rho_r \sin(\theta_s), \qquad 1 \leq r \leq R, \quad 1 \leq s \leq 4S. \tag{32}$$

The Maxwellian function $\mathcal{M}_j$ for the j^{th} equation of system (17) is then chosen as:

$$\mathcal{M}_j(u) = \frac{1}{J}\left(u + \frac{12rR}{v_{\max}(R+1)(2R+1)}\left(F^x(u)\cos(\theta_s) + F^y(u)\sin(\theta_s)\right)\right). \tag{33}$$

In [1] it is proven that for stability reasons one should choose $v_{\max}$ as follows:

$$v_{\max}^2 \geq \frac{12R^2\left(\|\partial_u F^x\|^2 + \|\partial_u F^y\|^2\right)}{(R+1)(2R+1)} \tag{34}$$

where $\|\cdot\|$ is the matrix norm associated with the classical 2-norm when $M > 1$.

3 Projective integration

The purpose of this paper is to construct a fully explicit, arbitrary order time integration method for the stiff system (17). The asymptotic-preserving property [11] implies that, in the limit when ϵ tends to zero, an ϵ-independent time step constraint, of the form $\Delta t = O(\Delta x)$, can be used, as the hyperbolic CFL constraint for the limiting equation (13). To achieve this, we will use a projective integration method [9, 17], which combines a few small time steps with a naive (*inner*) timestepping method, such as a direct forward Euler discretization, with a much larger (*projective, outer*) time step. The idea is sketched in figure 1.

The inner and outer integrator can be selected independently. In section 3.1, we discuss the inner integrator. Afterwards, in section 3.2, we consider the outer integrator, before studying numerically their properties in Section 4.

3.1 Inner integrators

We intend to integrate (17) on a uniform, constant in time, periodic spatial mesh with spacing Δx, consisting of I mesh points $x_i = i\Delta x$, $1 \leq i \leq I$, with $I\Delta x = 1$, and a uniform time mesh with time step δt, i.e., $t^k = k\delta t$. The numerical solution on this mesh is denoted as $\mathbf{f}^k_{i,j}$, where we have dropped the dependence on ϵ in the numerical solution for conciseness. After discretizing in space, we obtain a semi-discrete system of ordinary differential equations

$$\dot{\mathbf{f}} = \mathrm{D}_t(\mathbf{f}), \qquad \mathrm{D}_t(\mathbf{f}) := -\mathrm{D}_{\mathbf{x},\mathbf{v}}(\mathbf{f}) + \frac{1}{\epsilon}\left(\mathcal{M}_v(\mathbf{u}) - \mathbf{f}\right), \tag{35}$$

where $\mathrm{D}_{\mathbf{x},\mathbf{v}}(\cdot)$ represents a suitable discretization of the first order spatial derivative $v\partial_x$ (e.g., upwind differences).

As inner integrator, we choose an explicit scheme, for which we will, later on, use the shorthand notation

$$\mathbf{f}^{k+1} = S_{\delta t}(\mathbf{f}^k), \qquad k = 0, 1, \ldots \tag{36}$$

The forward Euler (FE) method and Runge-Kutta methods immediately come to mind.

Forward Euler (FE). The simplest time discretization routine is the forward Euler method,

$$\mathbf{f}^{k+1} = \mathbf{f}^k + \delta t\, \mathrm{D}_t(\mathbf{f}^k), \tag{37}$$

Higher-order Runge-Kutta methods. To obtain higher-order accuracy in time in the inner integrator, one could also employ any Runge-Kutta method [10, 28], such as the second order method

$$\mathbf{k}_1 = \mathrm{D}_t(\mathbf{f}^k), \tag{38}$$

$$\mathbf{k}_2 = \mathrm{D}_t\left(\mathbf{f}^k + \frac{\delta t}{2}\mathbf{k}_1\right), \tag{39}$$

$$\mathbf{f}^{k+1} = \mathbf{f}^k + \delta t\, \mathbf{k}_2. \tag{40}$$

How to choose the inner integrator will be discussed in section 4.

3.2 Outer integrators

In equation (17), the small parameter ϵ in the relaxation term leads to the classical time step restriction of the form $\delta t = O(\epsilon)$ for the inner integrator. However, as ϵ goes to 0, we obtain the limiting equation (19) for which a standard finite volume/forward Euler method only needs to satisfy a stability restriction of the form $\Delta t \leq C\Delta x$, with C a constant that depends on the specific choice of the scheme and the parameters of the equation.

In [17], it was proposed to use a projective integration method to accelerate such a brute-force integration; the idea, originating from [9], is the following. Starting from a computed numerical solution $\mathbf{f}^n$ at time $t^n = n\Delta t$, one first takes $K+1$ *inner* steps of size δt,

$$\mathbf{f}^{n,k+1} = S_{\delta t}(\mathbf{f}^{n,k}), \qquad 0 \leq k \leq K, \tag{41}$$

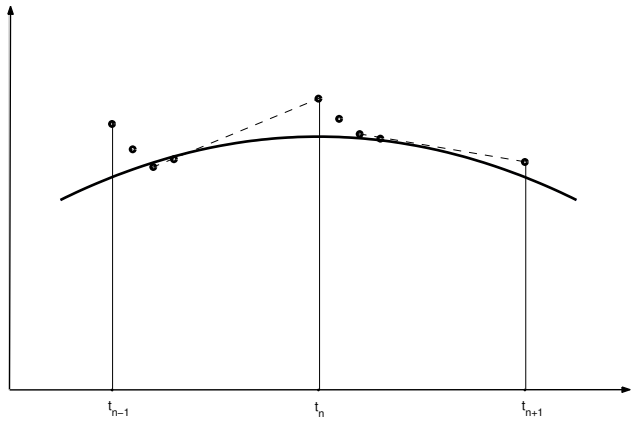

Figure 1: Sketch of the general idea of a projective integration method. At each time instance, an explicit method is applied over a number of small time steps so as to stably integrate the fast modes. As soon as these modes are sufficiently damped the solution is extrapolated using a much larger time step.

in which the superscript pair (n, k) represents the numerical solution at $t^{n,k} = n\Delta t + k\delta t$. The aim is to obtain a discrete derivative to be used in the *outer* step to compute $\mathbf{f}^{n+1} = \mathbf{f}^{n+1,0}$ via extrapolation in time, e.g.,

$$\mathbf{f}^{n+1} = \mathbf{f}^{n,K+1} + (\Delta t - (K+1)\delta t)\,\frac{\mathbf{f}^{n,K+1} - \mathbf{f}^{n,K}}{\delta t}. \tag{42}$$

This method is called projective forward Euler, and it is the simplest instantiation of this class of integration methods [9].

In [16], a higher-order projective integration method is constructed by replacing each time derivative evaluation $\mathbf{k}_s$ in a classical Runge-Kutta method by $K+1$ steps of an inner integrator as follows (with $\mathbf{f}^{n,0} = \mathbf{f}^n$ for consistency):

$$s = 1 : \begin{cases} \mathbf{f}^{n,k+1} & = \mathbf{f}^{n,k} + \delta t\ \mathrm{D}_t(\mathbf{f}^{n,k}), \qquad 0 \le k \le K \\ \mathbf{k}_1 & = \dfrac{\mathbf{f}^{n,K+1} - \mathbf{f}^{n,K}}{\delta t} \end{cases} \tag{43}$$

$$2 \le s \le S : \begin{cases} \mathbf{f}_s^{n+c_s,0} & = \mathbf{f}^{n,K+1} + (c_s\Delta t - (K+1)\delta t)\sum_{l=1}^{s-1}\dfrac{a_{s,l}}{c_s}\mathbf{k}_l, \\ \mathbf{f}_s^{n+c_s,k+1} & = \mathbf{f}_s^{n+c_s,k} + \delta t\ \mathrm{D}_t(\mathbf{f}_s^{n+c_s,k}), \qquad 0 \le k \le K \\ \mathbf{k}_s & = \dfrac{\mathbf{f}_s^{n+c_s,K+1} - \mathbf{f}_s^{n+c_s,K}}{\delta t} \end{cases} \tag{44}$$

$$\mathbf{f}^{n+1} = \mathbf{f}^{n,K+1} + (\Delta t - (K+1)\delta t)\sum_{s=1}^{S} b_s\mathbf{k}_s, \tag{45}$$

To ensure consistency, the RK matrix $\mathbf{a} = (a_{s,l})_{s,l=1}^{S}$, weights $\mathbf{b} = (b_s)_{s=1}^{S}$, and nodes $\mathbf{c} = (c_s)_{s=1}^{S}$ satisfy (see, e.g., [10]) the conditions $0 \le b_s \le 1$ and $0 \le c_s \le 1$, as well as

$$\sum_{s=1}^{S} b_s = 1, \qquad \sum_{l=1}^{S-1} a_{s,l} = c_s, \quad 1 \le s \le S. \tag{46}$$

(Note that these assumptions imply that $c_1 = 0$ by the convention that $\sum_1^0 \cdot = 0$.)

In the numerical experiments, we will specifically use projective Runge-Kutta methods of orders 2 and 4, represented by the Butcher tableaux in figure 2.

3.3 Stability of projective integration

We now briefly discuss the main stability properties of projective Runge-Kutta methods as derived in the companion paper [16]. To this end, we introduce the test equation and its corresponding inner integrator,

$$\dot{y} = \lambda y, \qquad y^{k+1} = \tau(\lambda\delta t)y^k, \qquad \lambda \in \mathbb{C}. \tag{47}$$

$\mathbf{c}$	$\mathbf{a}$
	$\mathbf{b}^T$

0		
$1/2$	$1/2$	
	0	1

0				
$1/2$	$1/2$			
$1/2$	0	$1/2$		
1	0	0	1	
	$1/6$	$1/3$	$1/3$	$1/6$

Figure 2: Butcher tableaux for Runge-Kutta methods. Left: general notation; middle: RK2 method (second order); right: RK4 method (fourth order).

As in [9], we call $\tau(\lambda\delta t)$ the *amplification factor* of the inner integrator. (For instance, if the inner integrator is the forward Euler scheme, we have $\tau(\lambda\delta t) = 1 + \lambda\delta t$.) The inner integrator is stable if $|\tau| \leq 1$. The question then is for which subset of these values the projective integration method is also stable.

Considering projective forward Euler, it can easily be seen from (42) that the projective forward Euler method is stable if

$$\left| \left[\left(\frac{\Delta t - (K+1)\delta t}{\delta t} + 1 \right) \tau - \frac{\Delta t - (K+1)\delta t}{\delta t} \right] \tau^K \right| \leq 1, \tag{48}$$

for all eigenvalues τ of the inner integrator for the kinetic equation (21). The goal is to take a projective time step $\Delta t = O(\Delta x)$, whereas $\delta t = O(\epsilon)$ necessarily to ensure stability of the inner brute-force forward Euler integration. Since we are interested in the limit $\epsilon \to 0$ for fixed Δx, we look at the limiting stability regions as $\Delta t/\delta t \to \infty$. In this regime, it is shown in [9] that the values τ for which the condition (48) is satisfied lie in the union of two separated disks $\mathcal{D}_1^{PFE} \cup \mathcal{D}_2^{PFE}$ where

$$\mathcal{D}_1^{PFE} = \mathcal{D}\left(1 - \frac{\delta t}{\Delta t}, \frac{\delta t}{\Delta t}\right) \text{ and } \mathcal{D}_2^{PFE} = \mathcal{D}\left(0, \left(\frac{\delta t}{\Delta t}\right)^{1/K}\right), \tag{49}$$

and $\mathcal{D}(\kappa, \mu)$ denotes the disk with center $(\kappa, 0)$ and radius μ. One then aims at positioning the eigenvalues that correspond to modes that are quickly damped by the time-stepper in $\mathcal{D}_2^{PFE}$, whereas the eigenvalues in $\mathcal{D}_1^{PFE}$ should correspond to slowly decaying modes. The projective integration method then allows for accurate integration of the modes in $\mathcal{D}_1^{PFE}$ while maintaining stability for the modes in $\mathcal{D}_2^{PFE}$.

We have the following result that compares the stability regions of higher-order projective Runge-Kutta methods to those of projective forward Euler in the limit when $\delta t/\Delta t$ tends to 0 [16].

Theorem 3.1 (Stability of higher-order projective Runge-Kutta methods) *Assume the inner integrator is stable, i.e., $|\tau| \leq 1$, and K and Δt are chosen in such a way that the projective forward Euler method is stable. Then, any projective Runge-Kutta method satisfying the conditions* (46)*, as well as the convexity condition*

$$0 \leq a_{s,l} \leq c_s, \qquad 1 \leq l \leq s, \;\; 1 \leq s \leq S, \tag{50}$$

is also stable.

Such a result is classical for regular Runge-Kutta methods. In [16], the proof is given in the projective Runge-Kutta case, revealing that the above property holds both for the stability domain corresponding to slow eigenvalues and for the stability domain corresponding to quickly damped eigenvalues.

Additionally, it is shown that in the limit when $\delta t/\Delta t$ tends to 0, the stability region breaks up into two regions $\mathcal{R}_1^{PRK}$ and $\mathcal{R}_2^{PRK}$ that satisfy

$$\mathcal{R}_1^{PRK,q} \supseteq \mathcal{R}_1^{PRK,q-1} \supseteq \mathcal{D}_1^{PFE} \text{ and } \mathcal{R}_2^{PRK,q} \supseteq \mathcal{R}_2^{PRK,q-1} \supseteq \mathcal{D}_2^{PFE}, \qquad \forall q,$$

in which the constant q indicates the order of the specific Runge-Kutta method. The proof relies on asymptotic expansions that are similar to (but much more tedious than) those mentioned in [9], see [16].

The main conclusion is that, whereas the stability regions of higher-order projective Runge-Kutta methods differ from those of projective forward Euler in their precise shape, their qualitative dependence on the parameters of projective integration (δt, K and Δt) is identical, and method parameters that are suitable for projective forward Euler will also be suitable for the higher-order projective Runge-Kutta method.